Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): A software package verification tool for verifying a software package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the tool comprising:

at least one test module configured to use the data entries of the file list to test at least one parameter of the software package;

a framework operable to identify the at least one test module defining a test of at least one parameter of the at least one software component of the package;

and

a control module operable to access the framework to cause the at least one test module identified therein to perform the test defined thereby for verifying the package.

Claim 2 (original): The tool of claim 1, wherein the framework identifies a plurality of test modules.

Claim 3 (original): The tool of claim 2, wherein the framework identifies a priority for each of the test modules.

Claim 4 (original): The tool of claim 3, wherein the control module is operable to cause the test modules to be executed sequentially according to the priority identified in the framework for each of the test modules.

Claim 5 (original): The tool of claim 1, wherein a mechanism is provided for identifying the at least one test module as being one of active and not active.

Claim 6 (original): The tool of claim 5, wherein the mechanism for identifying the at least one test module as being one of active and not active is included in the framework.

Claim 7 (original): The tool of claim 5, wherein the mechanism for identifying the at least one test module as being one of active and not active is included in the control module.

Claim 8 (original): The tool of claim 2, wherein the framework comprises a directory having a plurality of entries, each entry identifying one of the plurality of test modules.

Claim 9 (original): The tool of claim 8, wherein entry defines a priority for the one of the test modules identified therein.

Claim 10 (original). The tool of claim 8, wherein the identity of the one of the test modules defines its priority.

Claim 11 (original): The tool of claim 2, wherein each of the plurality of test modules is formed by a script and the framework identifies each of the test modules by a name for the script.

Claim 12 (original): The tool of claim 2, wherein each of the plurality of test modules is formed by a software object.

Claim 13 (currently amended): A computer program on a carrier medium for verifying a software package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the computer program comprising computer executable instructions:

a) providing at least one test module configured to use the data entries of the file list to test at least one parameter of the software package;

[[a)]] b) forming a framework operable to identify at least one test module for testing ~~defining a test of~~ at least one parameter of the at least one software component of the package;

and

[[b)]] c) forming a control module operable to access the framework to cause the at least one test module identified therein to perform the test ~~defined thereby~~ for verifying the package.

Claim 14 (currently amended): A program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform method steps for verifying a software package that includes at least one software component and wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the method comprising the steps of:

a) providing a framework for identifying at least one test module, each said test module defining a test of at least one parameter of the at least one software component of the package wherein said test is configured to use the data entries of the file list to test the at least one parameter of the software package;

b) accessing the framework to identify the at least one test module: and

c) causing the at least one test module to perform the test defined thereby on the package.

Claim 15 (original): The method of claim 14, wherein the framework identifies a plurality of test modules.

Claim 16 (original): The method of claim 15, wherein a priority for each of the test modules is identified in the framework.

Claim 17 (original): The method of claim 15, comprising sequentially causing each of the test modules to be executed according to the priority identified for each of the test modules.

Claim 18 (original): The method of claim 15, comprising identifying each of the test modules as being one of active and not active.

Claim 19 (original): The method of claim 15, comprising providing a directory in the framework, wherein the directory has a plurality of entries, each entry identifying one of the plurality of test modules.

Claim 20 (currently amended): A system for verifying a software package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the system comprising:

a) at least one test module configured to use the data entries of the file list to test at least one parameter of the software package;

[[a)]] b) a framework to identify at least one test module ~~defining a test of at least one parameter of the at least one software component of the package~~;

and

[[b)]] c) a control module operable to access the framework for causing the at least one test module identified therein to perform the test defined thereby for verifying the package.

Claim 21 (original): The system of claim 20, wherein the system comprises a computer including a processor, memory and software held in memory and operable to control the processor, the software forming:

said framework and said control module.

Claim 22 (currently amended): A computer system for verifying a software package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the system comprising:

a) a memory for storing software; and

b) a processing unit for executing the software to carry out the steps of:

(i) providing a framework to identify at least one test module defining a test of at least one parameter of the at least one software component of the package;

and

(ii) providing a control module operable to access the framework for causing the at least one test module identified therein to perform a test that uses the data entries of the file list to test the at least one parameter of the software package thereby ~~the test defined thereby for~~ verifying the package.

Claim 23 (currently amended): A method for verifying a software package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the method comprising the steps of:

a) providing a framework for identifying at least one test module, each said test module configured to use the data entries of the file list to test at least one parameter of the software package thereby defining a test of at least one parameter of the at least one software component of the package;

b) accessing the framework to identify the at least one test module; and

c) causing the at least one test module to perform the test defined thereby on the package.

Claim 24 (original): The method of claim 23, wherein the framework identifies a plurality of test modules.

Claim 25 (original): The method of Claim 24, wherein a priority for each of the test modules is identified in the framework.

Claim 26 (original): The method of claim 25, comprising sequentially causing each of the test modules to be executed according to the priority identified for each of the test modules.

Claim 27 (original): The method of claim 24, comprising identifying each of the test modules as being one of active and not active.

Claim 28 (original): The method of claim 24, comprising providing a directory in the framework, wherein the directory has a plurality of entries, each entry identifying one of the test modules.

Claim 29 (original): The method of claim 28, wherein each entry defines a priority of the test module identified thereby.

Claim 30 (original): The method of claim 28, wherein identity of a module defines its priority.

Claim 31 (currently amended): A method of verifying a software package that includes at least one software component, the method comprising the steps of:

a) receiving the software package wherein the software package includes a file list having data entries associated with parameters for the at least one software component;

b) accessing a framework that references at least one test module to identify the at least one test module from the framework, each said test module configured to use the data entries of the file list to define a test of the software package ~~defining a test of at least one parameter of the at least one software component of the package, for identifying the at least one test module from the framework~~; and

c) performing the test defined by the at least one test module on the package.

Claim 32 (currently amended): The method of Claim 31, including repeating steps (b) and (c) to perform a sequence of tests, the order in which the tests are performed being determined by relative priorities assigned to each of the at least one test module.

Claim 33 (original): A computer readable medium having stored thereon a data structure operable for us in verifying a software package that includes at least one software component, the data structure comprising:

a) a first field containing data representing one of a plurality of test modules, each test module being operable to test of at least one parameter of the at least one software component of the package,

b) where data representing ones of the test modules may be added to and deleted from the data structure, creating a flexible data structure.

Claim 34 (original): The medium according to claim 33, wherein the data structure further comprises a second field identifying a priority for each of the test modules represented by the data in the first data field, the priority defining an order of execution of test modules.

Claim 35 (original): The medium according to claim 33, wherein the data structure further comprises a third field identifying the one of a plurality of test modules represented by the data in the first data field as being one of active and not active.

Claim 36 (new): A software package verification tool as in Claim 1 wherein the software package is compliant with the SOLARIS standard.

Claim 37 (new): A software package verification tool as in Claim 36 wherein the file list comprises a "pkgmap" file.